Association for Information Systems
AIS Electronic Library (AISeL)

ECIS 2005 Proceedings

European Conference on Information Systems (ECIS)

2005

Exploring Cultural Issues in the Packaged Software Industry: A Usability Perspective

Maria Kutar
University of Salford, m.kutar@salford.ac.uk

Ben Light
University of Salford, b.light@salford.ac.uk

EXPLORING CULTURAL ISSUES IN THE PACKAGED SOFTWARE INDUSTRY: A USABILITY PERSPECTIVE

Kutar, Maria, IS, Organisations and Society Research Centre, University of Salford, Ashworth Building, Salford, M5 4WT, United Kingdom, m.kutar@salford.ac.uk

Light, Ben, IS, Organisations and Society Research Centre, University of Salford, Ashworth Building, Salford, M5 4WT, United Kingdom, b.light@salford.ac.uk

Abstract

In this exploratory paper, we analyse the packaged software industry, focussing specifically on usability issues associated with geographical/societal, sector and organisational cultures. We point to the different groupings that may constitute the 'communication' and 'task localisation' challenges associated with the roll out of globally standardised packages across and within diverse cultural settings. Our reading of the situation suggests several points of common interest between those engaged in Human Computer Interaction and Information Systems research. In particular: how the usability of products may be proffered in the processes of selection, implementation and use; usability based ethical considerations in the packaged software industry and contexts of use; the idea of 'best culture' and a wider conceptualisation of usability that goes beyond those features inscribed in the software. In conclusion, we argue that although only exploratory in nature, our work points to the value of this form of analysis in respect of other information and communication technologies. Moreover, we feel that our collaboration evidences the utility in fostering interdisciplinary relations between Human Computer Interaction and Information Systems.

Keywords: Packaged Software, Usability, Cultures, Human Computer Interaction.

1 INTRODUCTION

Within the Human Computer Interaction (HCI) community, cultures are often defined with reference to geographical boundaries. However, in addition to the geographical or societal, within Information Systems (IS), sector cultures and organisational cultures are prevalent considerations. In this paper, we will draw primarily upon the HCI and IS literatures to perform a brief deconstruction of the packaged software industry focussing specifically on cultural and product usability issues in the geographical/societal (macro), sector (meso) and organisational (micro) environments.

The packaged software[1] industry is global in nature (Brouthers and van't Kruis, 1997; Carmel, 1997; Houghton and Vickery, 2004) and as we shall explore further, for vendors, the benefits of selling packaged software derive largely from economies of scale. In order to attain these economies, products are standardised so they have broad appeal in the market (Fan et al., 2000; Sawyer, 2001). This allows vendors to develop one product, distribute the cost of development and ongoing maintenance over a large base and subsequently generate a profit (Gremillion, 1982; Dube, 1998; Butler, 1999). Clearly, the relatively recent astronomical rises in certain package application sectors (Enterprise Resource Planning and Customer Relationship Management for instance) implies widespread adoption but this does not automatically lead to the conclusion that these products are necessarily beneficial. Indeed, within IS, a growing number of studies are showing that end users often have problems appropriating generic packaged software products (Soh et al., 2000; Light, 2001; Pollock et al., 2003; Scott and Wagner, 2003). However, in the main, these studies tend to focus explicitly upon appropriation difficulties in terms of functionality or data misfits rather than the usability of the software[2].

Within the HCI community various principles and guidelines have been identified which aim to guide developers in building usable products. Recognising the diversity of users is a basic premise of appropriate design and usability principles require that interfaces can be adapted to user needs (Shneiderman, 1997). For example, one of Dix's three categories of such principles is 'flexibility', the multiplicity of ways the user and system exchange information (Dix et al., 2004). However, although it is argued that good interface design should cater for the needs of individual users, developers still have think in terms of categories of user, and have work on the basis of satisfying the majority of users. Thus, users as often grouped on the basis of influences such as their relative experience, their physical or intellectual capabilities, their age and, our focus here, their cultural background. Understanding that these influences contribute to the construction of differences in the way that a user will interact with software is an essential element of interface design, and enables evaluation of whether resulting needs have been met.

Therefore, there are 'global' packaged software products that are being appropriated in 'local' (culturally loaded) settings. Moreover, we have a body of thought in HCI which further advocates the recognition of differences amongst users. In the following sections, we bring these two issues together. We intend to explore the tensions in the packaged software industry with a specific focus upon culture and usability. We show that applying the principles of HCI, with their requirement that the cultural differences of users are catered for, poses particular challenges when systems are developed for a global market.

2 CULTURES AND PACKAGED SOFTWARE USABILITY

We are interested in the cultural issues that arise in the packaged software industry, but with a particular focus upon how these might play out with respect to the usability of the resultant software products. Culture is a slippery concept and surely, any definition will be subject to scrutiny. It can be seen as: a response to biological determinism; a delineator between what is viewed as civilised or barbarian; ideology in terms of a set of beliefs attitudes and opinions; artistic practices and ways of life (Abercrombie et al., 2000). In organisation studies, it has also been drawn upon to consider issues of 'usness' and 'themness' (Gabriel et al., 2000). For our purposes, we use the concept of culture to in cognisance of the ideas of usness and themness in the packaged software industry – vendors and consumers for instance and the issues this might raise as the groups involved carry out their 'ways of life'.

In terms of how cultural issues might play out with respect to software usability, there are two schools of thought. The first school argues that, making software usable, should focus upon whether the meaning of features such as symbols can be ascertained ((Bourges-Waldegg and Scrivener, 1998). For example, the colour red may represent danger (Anglo-American), life (India), happiness (China) and Royalty (France). The second school argues that whilst this is important, it is half of the story. This school, supported by Russo and Boor (1993); Del Galdo and Nielsen (1996) and Carey (1998), suggests that it is also important to consider the more contestable features which may be linked with such things as the ways of life the software will be used within. For example, Del Galdo and Nielsen (1996) distinguish between three levels of localization:

1. Displaying the native language, character set and notations.
2. Translating the user interface so that it is understandable and usable.
3. Matching the user's cultural characteristics, which goes beyond avoiding offensive icons and must accommodate the way business is conducted and that people communicate

Whilst levels 1 and 3 are distinct, level 2 considerations can be manifest within 1 and 3. Thus, we view the distinction as a duality based on issues of communication (level 1 and partial satisfaction of level 2) and task localization (levels 2 and 3). We define satisfactory communication as providing

language, character set, notations, icons and so on in a manner which the user can both understand and manipulate. It is important that in addition to being able to extract meaning, the user is also able to input information to the interface in an acceptable manner. We define task localization as providing the necessary means for the user to be able to carry out their task in a manner which matches their cultural characteristics.

In the following section we analyses the usability of packaged software products across the macro, meso and micro cultural environments paying particular attention to issues of communication and task localization.

3 MACRO CONSIDERATIONS

In this section we consider cultures at the geographical level, examining the issues of communication and task localisation within and across different countries, or regions. Communication, includes considerations such as translating text into the local language, and ensuring that local conventions are followed for the presentation of items such as time and date, currency and so on. In addition to the difficulties of translating between different alphabets, and writing systems, such translations may introduce communication problems which would not occur in the home country. Some writing systems read left to right (e.g. English), whilst others such as Hebrew and Arabic read from right to left. A particular difficulty with Hebrew and Arabic is that some technical terms have no direct translation and occasionally an English word will appear in parentheses. This requires that when entering text, a right to left insertion must switch to left to right and then back again, and that the appropriate character from the relevant alphabet is displayed. It has been noted that in the Hebrew version of Word 'it is possible to position the cursor such that the character displayed when you press a key depends on the direction of arrival of the cursor' (Ben-Ari, 1999). In terms of large configurable packages, these problems may occur at the data entry point. This example also illustrates the presence of 'us and them considerations' between vendors and consumers and amongst different consumers in the 'global' packaged software market.

An important issue in respect of task localisation is the question of whether the origin of a packaged software product relates to the inscribed (Akrich, 1992) assumptions about the conduct of work that they may display as a result. That is, if software is developed by people in the US – will it be in agreement with the 'ways of life' that exist in other countries? Indeed, it is suggested that the origin of the development of the package might be at odds with the country of implementation (Krumbholz et al., 2000). Of course, the country of origin of the software and the cultural composition of the development team may feed into the eventual product but this is not as simple as first suggested. As Carmel (1997) also points out, the US is in fact, an "intersection of cultures" and thus, for example, because Indian people may work as developers in the US, it is difficult to pinpoint 'which' cultures get inscribed in the software. Thus, in terms of cultures, it is perhaps more useful to consider the geographical argument in relation to the problems of 'us' implementing a generic product that has been targeted at a broad market by 'them'. Undoubtedly, there will be features of the product that will be engrained with cultures that may be influenced by the origins of production. The issue in relation to usability appears to be understanding the implications of this for the context within which the product will be used.

4 MESO CONSIDERATIONS

The meso, sector, environment can also be said to be comprised of diverse cultures. At the communication level, this requires that software uses language and symbols which reflects the domain in which it is intended to be used. It is increasingly common for modules within a software package

suite to be re-badged for specific industries, often with minimal development for that context. For example it is well known that the SAP product was originally built for the chemical process industries. However, now it is sold as a product that has been targeted at, and in existence for thirty years in no less than 25 different industries (SAP, 2003). Yet, as noted elsewhere:

> "Today, most ERP packages come in different industry specific "flavours," but in some cases the degree of fit may still be low" (Brehm et al., 2001, p. 2)

Moreover, there are studies which indicate problems with the standardised terminology and formatting of reporting capabilities embedded in the software. For example, in one ERP project, the company reported that the documentation provided by the consulting group was not tailored to their needs. For example, a costing invoice was called a 'different outlet' which did not make much sense to their employees (Skok and Legge, 2001). Unit of measurement is another potential sector specific communication problem. In one study, a representative of a chemical company identified a problem in their ERP implementation. They reported that the stock levels in their sector fluctuated depending upon the temperature because their products were gaseous[3].

At the task localisation level, great sensitivity is required in order to provide packages which can be said to be culturally sensitive to the needs of the user in any given industry. For example, at Cable, there was the culture of the 'book' whereby it was accepted practice for customers to reserve a quantity of copper at a set price as it was subject to market fluctuations in the same way as oil (Light, 2001). Initially, the global product did not account for the industry specific level of task localisation. Thus, we can see a tension. In the end, the vendor incorporated this functionality into the generic product for the industry, because THEY could see demand for it. However, other customisations were also performed to make the product more usable that were not treated by the vendor in the same fashion. Fundamentally, the vendor was in an untenable position vis-à-vis their business model of selling a generic product, to as wide a market as possible.

5 MICRO CONSIDERATIONS

Finally, there are communication and task localisation issues to consider at the micro, organisational level. Here it is important to remember that we see organisations as also part of one or a number of meso and macro environments. Thus, the issues we have already discussed will, of course, apply here at the micro level. Moreover, even at the organisational level, we recognise further diversity as we do not adopt a unitary view. We take organisations to be comprised of subcultures, rather than having 'a culture' (Gabriel et al., 2000). In terms of communication, terminology is something that is a contested and very particular feature of the diverse cultures making up organisations from within and in differentiating organisations in the same industry or geographic locale. The following quote extracted from a case of the development of a university enterprise system product illustrates this very well:

> "Supplier Analyst: Students with bad marks. What do you do with them, leave them in limbo or give them a second chance?
>
> Southern University: Depends on timing, if just before a session and there is no chance of them bettering their mark, then we refuse them. Or, alternatively, we could say we've not decided yet. That is not a hold but a 'waiting status'.
>
> Technology University: If you're doing something that might pick up your grades?
>
> Supplier Analyst: I wouldn't call that a hold, that's a 'provisional situation'.
>
> Rural University: We have a 'partial hold', so holds affects some things ….
>
> Large Campus University: Isn't that a 'half-hold' …"
>
> (Pollock et al., 2003, p. 326)

Unsurprisingly, the difficulties are further amplified when task localisation is considered. At 'Ivy' the way that Principal Investigator (PI) report generation had to be enacted within the new package was deemed to be so cumbersome and inefficient that a what were termed 'shadow systems' emerged (Scott and Wagner, 2003). Indeed, these customisations gained ground throughout Ivy as a way of making the generic product more usable in the local setting with the result being that the wider system that the PI was part of is now being developed into a full scale alternative to the original generic package. This is not an isolated incident either. The development of the Despatch customisation at 'Home' had much the same history (Light, 2001). The generic package required the entry of the same data numerous times and thus, keystroke emulation software was used to automate the processes embedded in the software to make it more usable.

6 DISCUSSION AND IMPLICATIONS

HCI studies usually take a geographical approach to cultural issues. Of course this is valuable as it often highlights the more prominent usability issues. As our work demonstrates, geographical communication and task localisation aspects of the usability of packaged software are important considerations. Yet, the packaged software industry has to operate a geo-cultural homogenization strategy as much as possible in order to achieve economies of scale. Thus, we argue that attention to industry and organisational cultures are also necessary from a usability perspective. The packaged software industry illustrates this problematic. Whilst vendors tout industry targeted suites of products, they have been reported as lacking in communication and task localisation aspects, and this is further reported on at the organisational level. A brief summary of these potential problems is presented in Table 1.

Table 1: Exemplar Cultural Conisderations in the Packaged Software Industry

Environments / Cultural Considerations		Macro	Meso	Micro
Potential 'usness' and 'themness' groupings		• Vendors-Consumers • Global consumers	• Vendors-Consumers • Sector consumers	• Vendors-Consumers • Local consumers
Ways of life considerations	Communication exemplars	• Geographically specific language • Currency • Date format • Colour symbolism	• Sector specific language • Unit of measurement (barrel v bottle)	• Organisation specific language • Input conditions (office v building site)
	Task localisation exemplars	• Geo-cultural inscriptions	• Sector-cultural inscriptions	• Organisational and sub-cultural inscriptions

Within information systems, the consideration of usability issues if often not of prime importance. Again though, we suggest there is utility in thinking about the fit of software beyond functionality. For example, communication influences the look and feel of products, a key component of vendor sales pitches whom those in an implementing organisation are subject to in packaged software

selection environments (Howcroft and Light, 2002). In implementation, the communication and task localisation characteristics of a piece of software may, for instance, play a part in user acceptance and the levels of training required. Ultimately, these features can also influence the effectiveness with which these packages are used. For example, Cadbury Schweppes implemented plans to fulfil 250 orders where normally they would fulfil 1000 due to the increased complexity and the need to re-train staff post SAP implementation (August, 1999).

For clarity, we have organised the cultural issues within each of the environments. However, the micro environment is part of a meso environment, which in turn is part of a macro environment. Thus, geo-cultural issues are manifest at the micro level. Moreover, further complications arise when we consider that those in organisations, at the micro level, might 'share' information systems with those in organisations in other industries, in other geographic locales. Soh et al. (2000) provide the example of an ERP package aimed at the healthcare industry. The billing function proved unsuited to Singapore's healthcare industry model, where payments are made by the individual to the hospital (often in instalments), in contrast to many Western healthcare industry models whereby governments or insurance companies deal with payment for care on behalf of the patient. This necessitates very different 'task localisation' considerations.

At the heart of the packaged software industry is the problematic of the need to sell global products to fit local settings. As reported elsewhere, this represents something of an ethical dilemma (Adam and Light, 2004). In terms of our focus here, thought needs to be given as to what can be expected of vendors in terms of doing the right thing vis-à-vis developing usable products for its oft global consumer base. Moreover, there is a further problem in that, with respect to cultures, we would argue that vendors need to unpack the 'local' in at least three ways in line with our deconstruction. Yet, even if vendors were to attempt this, it is clear that attempting to cover many cultures, for example as with English v Arabic language, can lead to usability difficulties for the user.

In addition to ethical considerations for groups such as vendors, the latter discussion, suggests that users also have to consider their position. In order to explain this we first have to unpack the idea of the user. For brevity here, we shall take users to be comprised of two groups operational staff and management staff (clearly this is simplistic). Ethically, operational and management staff, it is arguable, both need to be respectful of cultural diversity. Thus, although it might be more difficult to work with a culturally sensitive system, it might be the right thing to do. However, we are not so naïve to assume that this is necessarily achievable. Information systems are usually introduced into organisations on the managerially oriented grounds of competitiveness and efficiency rather than any vision for doing good. Thus, operational staff are likely to accept managerial prerogative and managers – strive for features that meet their aims. Thus, we argue, that managers – the usual representatives in packaged software product development initiatives, are more likely to conspire, unwittingly or otherwise, with packaged software vendors, to build products that are standardised to the extent that they are as culturally homogenous and as multi-culturally insensitive as possible.

This then raises further dilemmas in terms of usability. What is deemed to be 'best practice' within or 'best culture' components of a given culture and thus worthy of attention in the product development/selection effort. Again, this is something usually constructed by vendors and managers. For example, the product enhancement process of the SAP Treasury module illustrates how exclusive, and vendor controlled, the packaged software development process can become (Scott and Kaindl, 2000):

- Most of those in the participating organisations were managers from large global firms.
- The SAP team selected participants with 'state of the art' knowledge in the area.
- To 'ensure mutual goal alignment', they chose organisations that were willing to change their processes.
- The chosen individuals felt like members of an 'elite group', were excited to influence the design and to be among the first customers to have the module.

Taking a more geo-cultural perspective, there are further considerations for those in countries not involved in the initial development of products. As Walsham points out more generally:

> "the foreigner in any country needs to be constantly vigilant when thinking that difference implies inferiority. The worst examples of such an attitude are often provided by people from Western countries who mistakenly equate high economic living standards with high cultural and ethical standards. The business world often tends to reinforce such values, whereas cultural sensitivity implies the need to see economics as only one aspect of life." (Walsham, 2001, p. 202-203)

In recent years, it has become clearer that large configurable packages cannot be implemented and used 'out of the box'. Arguably, this is because packages not only have processes inscribed in them that make them appear more or less usable, they imply processes that are more or less usable too (Lee and Lee, 2000). Thus, there is the question of whether HCI studies need to stretch to consider the context of use more deeply. Do developers responsible for the usability of software need to be sensitive to the cultural issues surrounding the usability of the processes implied by the software as well as those embodied by it. Clearly, throughout the packaged software industry, many vendors/consultants have sold the idea of adopting generic packages 'out of the box' and thus users have implemented products which expose tensions vis-à-vis usability. In response, the customisation of packages is becoming more popular (or at least more widely admitted and publicised) in order to make products more usable. The question again therefore arises is this something to be expected and accepted with this form of systems development, even though it moves the burden of development back to the user base, when at this point in time, generic packages are usually adopted to outsource this?

7 CONCLUSION

In this exploratory paper, we briefly unpack the packaged software industry from a cultural usability perspective. Through this example, we point to how these considerations unearth interesting issues for IS and HCI research and practice. The packaged software industry illustration provides an excellent example as it is based upon global–local tensions. Vendors have to sell culturally loaded products across multicultural contexts. However, even if vendors work towards recognising cultural diversity, there may still be problems as users struggle themselves in working with the end result. Moreover, these issues are not specific to the packaged software industry. They apply equally well to other 'global' information and communications technologies such as mobile phones, and even other forms of systems development projects, such as those based on open source or custom development technologies that cut across diverse cultures. As is already recognised in terms of functionality-fit, within information systems, there are also questions surrounding how products are domesticated to make them usable. Although these questions have received some implicit and explicit attention, we believe they deserve more and, indeed, should continue to prove fruitful areas of investigation for IS and HCI researchers alike.

8 REFERENCES

Abercrombie, N., Hill, S. and Turner, B. S. (2000), *The Penguin Dictionary of Sociology*, 4th edn., Penguin Books, London.

Adam, A. and Light, B. (2004), "Selling Packaged Software: An Ethical Analysis", in *Proceedings of the 12th European Conference on Information Systems* Turku, Finland, apologies the remainder of this reference is available upon request.

Akrich, M. (1992), "The De-Scription of Technical Objects", in Bijker, W. E. and Law, J. (Eds), *Shaping Technology/Building Society: Studies in Sociotechnical Change*, MIT Press, London, pp. 205-224.

August, V. (1999), "ERP Sites Hit by Performance Dip", *Information Week*, 17 February, p. 12.

Ben-Ari, M. (1999), "Bricolage Forever", in *Proceedings of the 11th Annual Workshop of the Psychology of Programming Interest Group* Leeds, UK.

Bourges-Waldegg, P. and Scrivener, S. A. R. (1998), " Meaning, The Central Issue in Cross-Cultural HCI Design", *Interacting with Computers*, 9, pp. 287-309.

Brehm, L., Heinzl, A. and Markus, M. L. (2001), "Tailoring ERP Systems: A Spectrum of Choices and their Implications", in *Proceedings of the 34th Hawaii International Conference on System Sciences* IEEE Press: Maui, Hawaii, pp. CD-ROM.

Brouthers, K. D. and van't Kruis, Y. M. (1997), "Competing in Software: Strategies for Europe's Niche Businesses", *Long Range Planning*, 30(4), pp. 518-528.

Butler, J. (1999), "Risk Management Skills Needed in a Packaged Software Environment", *Information Systems Management*, 16(3), pp. 15-20.

Carey, J. M. (1998), "Creating Global Software: A Conspectus and Review", *Interacting with Computers*, 9, pp. 449-465.

Carmel, E. (1997), "American Hegemony in Packaged Software Trade and the "Culture of Software"", *The Information Society*, 13(1), pp. 125-142.

Del Galdo, E. and Nielsen, J. (1996), *International User Interfaces*, John Wiley and Sons, London.

Dix, A., Finlay, J., Abowd, G. D. and Beale, R. (2004), *Human-Computer Interaction*, 3rd edn., Prentice Hall.

Dube, L. (1998), "Teams in Packaged Software Development: The Software Corp. Experience", *Information Technology and People*, 11(1), pp. 36-61.

Fan, M., Stallaert, J. and Whinston, A. B. (2000), "The Adoption and Design Methodologies of Component-Based Enterprise Systems", *European Journal of Information Systems*, 9(1), pp. 25-35.

Gabriel, Y., Fineman, S. and Sims, D. (2000), *Organizing and Organizations*, 2nd edn., Sage Publications, London.

Gremillion, L. L. (1982), "Improving Productivity with Application Software Packages", *Business Horizons*, 25(2), pp. 51-54.

Houghton, J. W. and Vickery, G. (2004), *Digital Delivery of Business Services*, Organisation for Economic Co-operation and Development, Paris.

Howcroft, D. and Light, B. (2002), "A Study of User Involvement in Packaged Software Selection", in Applegate, L., Galliers, R. D. and De Gross, J. I. (Eds), *Proceedings of the 23rd International Conference on Information Systems* Association for Information Systems: Barcelona, Spain, pp. 69-77.

Krumbholz, M., Galliers, J. and Coulianos, N. (2000), "Implementing Enterprise Resource Planning Packages in Different Corporate and National Cultures", *Journal of Information Technology*, 15(4), pp. 267-279.

Lee, Z. and Lee, J. (2000), "An ERP Implementation Case Study from a Knowledge Transfer Perspective", *Journal of Information Technology*, 15(4), pp. 281-288.

Light, B. (2001), "The Maintenance Implications of the Customization of ERP Software", *The Journal of Software Maintenance: Research and Practice*, 13(6), pp. 415-430.

Pollock, N., Williams, R. and Procter, R. (2003), "Fitting Standard Software Packages to Non-Standard Organizations: The 'Biography' of an Enterprise-Wide System", *Technology Analysis and Strategic Management*, 15(3), pp. 317-332.

Russo, P. and Boor, S. (1993), "How Fluent is Your Interface? Designing for International Users." in Ashlund, S., Mullet, K., Henderson, A., Hollnagel, E., & White, T. ((Ed.) *Proceedings of INTERCHI '93 Conference on Human Computing Systems, INTERACT '93 and CHI '93* ACM Press, New York: Amsterdam, pp. 342-347.

SAP (2003), *Designed For Your Industry, Scaled To Your Business, Ready for Your Future*, SAP, Newtown Square, PA.

Sawyer, S. (2001), "A Market-Based Perspective on Information Systems Development", *Communications of the Association for Computing Machinery*, 44(11), pp. 97-102.

Scott, J., E. and Kaindl, L. (2000), "Enhancing Functionality in an Enterprise Software Package", *Information and Management*, 37(3), pp. 111-122.

Scott, S. V. and Wagner, E. L. (2003), "Networks, Negotiations, and New Times: The Implementation of Enterprise Resource Planning into an Academic Administration", *Information and Organization*, 13(4), pp. 285-313.

Shneiderman, B. (1997), *Designing the User Interface: Strategies for Effective Human Computer Interaction.*, Addison-Wesley, Boston.

Skok, W. and Legge, M. (2001), "Evaluating Enterprise Resource Planning (ERP) Systems Using an Interpretive Approach", in Serva, M. (Ed.) *Proceedings of the ACM SIGCPR Conference on Computer Personnel Research* ACM Press: Sandiego, USA, pp. 189-197.

Soh, C., Siew Kien, S. and Tay-Yap, J. (2000), "Cultural Fits and Misfits: Is ERP a Universal Solution?" *Communications of the Association for Computing Machinery*, 43(4), pp. 47-51.

Walsham, G. (2001), *Making a World of Difference: IT in a Global Context*, John Wiley and Sons Ltd, Chichester.

---

[1] Of course packaged software can take many forms, is developed for different contexts and can be commercially and non-commercially licensed. In this paper we focus upon the work organisation and larger configurable packages (such as that for workflow or enterprise support) that are commercially licensed. However, clearly, there will be issues that we raise that will have value for domestic usage of packages, other forms of packaged software and that which is not commercially licensed.

[2] Although Soh et al. (2000) identify difficulties which we argue are based on usability rather than purely functionality.

[3] Unpublished interview data from Holland and Light (1997-1999) EPSRC Project: Generic Systems Design Strategies to Overcome the Problems of Legacy Systems, Manchester Business School, University of Manchester.